



**The University of Sydney**

**School of  
Information Technologies**

**COMP5348**

**Enterprise-Scale Software Architecture**

**6 Credit Points**

**Unit of Study Outline & Assessment Details**

**Semester 1, 2009**

**IMPORTANT: Policy relating to Academic Dishonesty and Plagiarism.**

The School of Information Technologies<sup>1</sup> views all forms of academic dishonesty, including plagiarism and recycling, very seriously.

*Plagiarism* means presenting another person's ideas, findings or work as one's own by copying or reproducing them without due acknowledgement of the source.

*Recycling* means the submission for assessment of one's own work, or of work which is substantially the same, which has previously been counted towards the satisfactory completion of another unit of study, and credited towards a university degree, and where the examiner has not been informed that the student has already received credit for that work.

Students who submit work containing significant portions that have been copied from other sources, including published works, the internet, existing programs, work previously submitted for other awards or assessments, or the work of other students, without proper acknowledgement will be penalised. Decisions as to the penalty may include:

- (a) counselling the student;
- (b) issuing a written warning;
- (c) requiring the student to resubmit the work for assessment; or to undertake other remedial work;
- (d) requiring the student to undertake another form of assessment in lieu of the assignment in question, such as an unseen examination;
- (e) applying a fail grade to the work, or part thereof, submitted for assessment;
- (f) applying a fail grade overall in the unit of study; or
- (g) referring the matter to the Registrar if the head of school considers there has been a breach of the University's standards of academic honesty and the student continues in a denial, or, following the interview, the head of school considers that failing the unit of study is insufficient to deal with the matter.

Where there is doubt about which portions of work are contributed by a particular student he/she may be required to demonstrate knowledge of the relevant material by answering oral questions or by undertaking supplementary work, either written or in the laboratory, in order to arrive at the final assessment mark.

---

<sup>1</sup> Refer to Academic Board policy: [http://www.usyd.edu.au/ab/policies/Academic\\_Honesty\\_Cwk.pdf](http://www.usyd.edu.au/ab/policies/Academic_Honesty_Cwk.pdf)

# COMP5348 Enterprise-Scale Software Architecture

## 1. Introduction

This unit covers topics on software architecture for large-scale enterprises. Computer systems for large-scale enterprises handle critical business processes, interact with computer systems of other organisations, and have to be highly reliable, available and scalable. This class of systems are built up from several application components, incorporating existing "legacy" code and data stores as well as linking these through middleware technologies, such as distributed transaction monitors, remote objects, message-queuing, publish-subscribe, and clustered application servers. The choice of middleware can decide whether the system achieves essential non-functional requirements such as performance and availability. The objective of this unit of study is to educate students for their later professional career and it covers Software Architecture topics of the ACM/IEEE Software Engineering curriculum.

## 2. Objectives

This subject will cover both fundamental concepts that apply across many technologies, and practical work with specific technologies.

Students who successfully complete this unit will have:

1. Understand the role of a software architect; Respect what he/she is doing, and why; Know when to involve him/her. (Graduate Attribute Type : Professional Practice )
2. Demonstrate broad knowledge of common architectural approaches for enterprise software, as well as detailed skills with some technologies available to implement those approaches. (Graduate Attribute Type : Discipline Specific Expertise )
3. Produce written evaluation of different architectures and/or of different technologies. (Graduate Attribute Type : Communication )
4. Knowledge of fundamentals of performance analysis, performance measurement principles, and state management in face of concurrency and distribution. (Graduate Attribute Type : Science and Engineering Fundamentals )
5. Work in small teams of people with diverse skills and backgrounds, to complete demanding tasks which are loosely defined and require rapid learning of new concepts. (Graduate Attribute Type : Teamwork and Project Management )

## 3. Entry Requirements

Programming competence in Java or a similar OO language. Capacity to master novel technologies (especially to program against novel APIs) using manuals, tutorial examples, etc.

## 4. Unit of Study Delivery

A variety of learning situations will be employed during the unit of study, including lectures, on-line demos, prescribed reading, tutorials, pen and paper exercises, and directed computer laboratory exercises. To benefit fully from this unit it is necessary to participate fully in all aspects of the unit of study.

## 5. Expectations

1. Students are expected to attend all scheduled lectures, and laboratory classes. You should expect to spend a minimum of nine hours per week including scheduled lectures and laboratory times.
2. Students are expected to undertake prescribed reading, to carry out exercises and laboratory tasks and to submit selected work for assessment as directed. It should be realised that laboratory exercises are expected to take longer than just the time scheduled for classes.
3. Students are expected to self-dependently prepare the prescribed research paper readings and conduct additional literature and system research as necessary.
4. Students are expected to be able to work independently and to make effective use of a range of resources including the library, the Internet and relevant on-line help facilities.
5. Students are expected to check their progressive results regularly. Results will be published through

*USyd eLearning*. Any errors or omissions must be reported to the unit coordinator, with appropriate evidence, as soon as possible. **Please note:** Marks are considered to have been confirmed ten days after being published and will not subsequently be altered.

## 6. Assessment Package

<i>Component</i>	<i>Week</i>	<i>Learning Outcomes Assessed</i>	<i>% of Final Grade</i>
Assessment 1: Group Project	6	2,3,5	10
Assessment 2: Group Questions	8	2,4	10
Assessment 3: Group Project	11	2,3,5	10
Assessment 4: Group Questions	12	1,2,3,4	10
Final examination, 2 hour paper		1,2,3,4	60

***It is a policy of the School of Information Technologies that in order to pass this unit, a student must achieve at least 40% in the written examination. A student must also achieve an overall final mark of 50 or more. Any student not meeting these requirements can achieve a maximum mark of no more than 45.***

**Late work:** In the interests of fairness to all students, the School of Information Technologies policy states that late work cannot be accepted. In exceptional cases late work must be submitted *directly to the unit of study coordinator* accompanied by an application for Special Consideration as outlined at [http://www.it.usyd.edu.au/current\\_students/postgrad\\_coursework/policies/specons.shtml](http://www.it.usyd.edu.au/current_students/postgrad_coursework/policies/specons.shtml)

Assessment results will be published through *USyd eLearning*. Students are required to check their results.

Any errors or omissions must be reported to the unit coordinator, with appropriate evidence, within ten (10) days of being published. Ten days after being published, marks are considered to have been confirmed and will not subsequently be altered.

## 7. Details of Assessment Components

### 7.1 Assessment 1: Group Project

In this practical assignment, students work in groups using technologies for distributed computing. Due in week 6, worth 10%.

### 7.2 Assessment 2: Group Questions

In this paper-based assignment, students work in groups to answer questions on material of weeks 1-6. These tasks are similar to questions on the final exam. Due in week 8, worth 10%.

### 7.3 Assessment 3: Group Project

In this practical assignment, students work in groups using asynchronous communication technologies. Due in week 11, worth 10%.

### 7.4 Assessment 4: Group Questions

In this paper-based assignment, students work in groups to answer questions on material of weeks 7-11, and also reflecting on practical experiences in assignments and labs. These tasks are similar to questions on the final exam. Due in week 11, worth 10%.

### 7.5 Written Examination

The written examination will cover all aspects of the unit of study. It will test the candidates' ability to discuss issues critically and to apply the knowledge learnt during the course them to specific situations. Duration of the examination will be two hours. Note the School of IT policy on special consideration for examinations described at [http://www.it.usyd.edu.au/current\\_students/postgrad\\_coursework/policies/specons.shtml](http://www.it.usyd.edu.au/current_students/postgrad_coursework/policies/specons.shtml)

## 8. Teaching team

This unit of study will be coordinated and lectured by:

	<i>Room</i>	<i>Phone</i>	<i>Consultation time</i>	<i>Email</i>
Dr Alan Fekete (coordinator and some lectures)	SIT 4E-447	93514287	Mon 5pm	fekete@it.usyd.edu.au
Dr Uwe Roehm (some lectures)	SIT 4E-446	90365305	Mon 2pm	roehm@it.usyd.edu.au

Note that alternative consultation times can be arranged as needed; just email the lecturer.

## 9. Textbook and Readings

The main reference for much of the course will be:

Essential software architecture / Ian Gorton, 005.12 61

Other books with useful material for parts of the course include:

Web services : concepts, architectures and applications / Gustavo Alonso ... [et al.]. 005.72 143

Performance by design : computer capacity planning by example / Daniel A. Menascé, Virgilio A. F. Almeida, Lawrence W. Dowdy. 004.24 22

All these reference books are available at the SciTech library on reserve, for short-period use by comp5348 students.

### 9.1 Course Web Page

For most recent information on this course, lecture slides, exercises etc. visit USyd eLearning or the course's web page at <http://www.it.usyd.edu.au/~comp5348/>

## 10. Lecture and Assessment Schedule

This schedule is indicative only - the list of topics and dates may change. Lecture slides are usually posted on the unit website a few days after the lecture.

Wk	Lecture	Topic	Assessment due
1	Thu 5 March	Introduction; Definition of the field; Nature of Software Architect's Role; Types of non-functional requirements ("ilities")	
2	Thu 12 March	State and Concurrency	
3	Thu 19 March	Distributed Computing; RPC, Distributed Objects, and Component Models	
4	Thu 26 March	Distributed Transactions	
5	Thu 2 April	Performance Measures	
6	Thu 9 April	Predicting performance	Assessment 1
7	Thu 23 April	Asynchronous Messaging and related technologies I	
8	Thu 30 April	Asynchronous Messaging and related technologies II	Assessment 2
9	Thu 7 May	Service Oriented Architectures	
10	Thu 14 May	Fault-tolerance concepts	
11	Thur 21 May	Replication for scalability and availability	Assessment 3
12	Thu 28 May	Guest lecture?	Assessment 4
13	Thu 4 June	Unit Review	